

REMARKS

File History

The latest Office action of 5/22/2006 comes after Applicant's traverse of restrictions dated 2/23/2006.

In a brief telephone exchange with SPE Lee on July 24, 2006 it was confirmed that claims not indicated as being presented are finally withdrawn pursuant 35 USC §121. (The Primary Examiner was unavailable through end of July. SPE Lee pointed to summary paragraph 4a on the cover sheet as finally disposing of the restricted out claims.)

The following allowances, rejections, objections and other actions appear to have been made in the latest action therefore:

- Claims 1, 41-44 were rejected under 35 USC §103/102(e) as being obvious in view of LeClair (US 6,636,891, issued 10/21/2003 but based on an application filed Nov. 6, 1998) in combination with Phaal (US 6,055,564, issued 4/25/2000).
- Claims 4 and 17-20 were rejected for indefiniteness pursuant to 35 USC §112.
- Claims 3, 5-16, 21, 36-40 were allowed.
- Claims 2, 22-35 were deemed as finally withdrawn.

Summary of Current Response

Claims 4, 41-44 are amended.

Claims 2, 22-35 are canceled without prejudice.

Arguments are presented concerning the applied art and its proposed combination.

Applicants' Overview of Outstanding Office Action

Applicant sees the outstanding Office action of 5/22/2005 as having the following noteworthy feature:

- (1) In rejecting Claims 1, 41-44 over new art (LeClair combined with Phaal), the PTO makes inaccurate findings of fact regarding the teachings of each of these references. Neither teaches or suggests that portion of Claim 1 for example, which recites "(b) in response to receipt of the first time stamp, sending from the job requestor to the job processor, a combination of job payload data and a second time stamp also representing the scheduled, first time;" [*Emphasis added.*]

The alleged errors of fact finding are detailed below. It is respectfully submitted that a prima facie case of unpatentability has not been made out.

Ordinary construing of language of Claim 1

Paragraph (a) of Claim 1 recites: "(a) issuing to the job requestor, a first time stamp representing a respectively scheduled, first time within a timing reference frame of the job processor at which a respective first job is to be performed;" [*Emphasis added.*]

The PTO asserts that **LeClair col. 8, lines 16-22** fully meets all aspects of this element (a) of Claim 1, in other words, that LeClair's server (e.g., 640 of Fig. 6) issues to the initiator/requestor (e.g., 600) something that qualifies as a first time stamp representing a respectively scheduled, first time within a timing reference frame of the server (640) at which a respective first job (e.g., print out of bulk data by the printer 780 of Fig. 7) is to be performed.

Applicant respectfully submits that this is incorrect and that a fair reading of LeClair col. 8, lines 16-22 does not provide any such teaching --as will be detailed below. There is no "first time stamp" in LeClair and there is no scheduling of when the "job" (e.g., printing) will be performed.

Paragraph (b) of Claim 1 recites: "(b) in response to receipt of the first time stamp, sending from the job requestor to the job processor, a combination of job payload data and a second time stamp also representing the scheduled, first time;" [*Emphasis added*].

The PTO asserts in a first instance (OA page 3) that **LeClair col. 8, lines 16-22, 37-40** fully meets all aspects of this element (b) of Claim 1, in other words, that there is a sending in LeClair's system of a "combination" of a second time stamp and job payload data (e.g., where the payload is the bulk data that is to be printed and was temporarily stored in 620 or 622 of LeClair Fig. 6).

The PTO contradictingly admits in a second instance (OA page 4) that LeClair does not teach such a "combination" of a second time stamp and job payload data. (The PTO argues that Phaal fills this admitted deficiency.)

Additionally at OA page 4, the PTO action discusses something about not being able to perform at the time of the first stamp and instead performing at the time of the second stamp. Applicant is confused by this and respectfully submits that Claim 1 does not set forth such an operation. Claim 1 does not say that the second time stamp has to be different than the first time stamp. Instead, Claim 1 states that the "(b) ... second time stamp also represent[s] the scheduled, first time;" [*Emphasis added, text modified for clarity*].

It appears from the immediate above item that the PTO is rejecting subject matter that is different from what the plain words of Claim 1 recite. It appears that the PTO is projecting Phaal's teachings into Claim 1 rather than reading Claim 1 for what it says on its own and in view of the specification. Reconsideration is respectfully requested.

Paragraph (d) of Claim 1 recites: "(d) causing the job processor to process the stored payload data when a time corresponding to the second time stamp occurs ..." (where pursuant to paragraph (b), the second time stamp also represent[s] the scheduled, first time) [*Emphasis added*].

In LeClair, the payload data is not yet in the server at the time the request is considered for servicing (see paragraph (c) of Claim 1). It is understood how the PTO might have been led into the thinking the second time stamp of Claim 1 might represent some very different second time, but upon closer inspection of Claim 1 it should be apparent that both of the first and second time stamps generally represent a same pre-scheduled time within the

timing reference frame of the job processor. For purpose of literal accuracy, paragraph (d) identifies it as the second time stamp, because it is the second time stamp signal that the job processor responds to. (But see also the caveat of the next paragraph.)

(Although not literally stated in Claim 1, it would be within the spirit of the claimed invention and under the doctrine of equivalents to add a constant offset to all second time stamps so as to trick the job processor into performing all its scheduled jobs at a constantly offsetted time in the timing reference of the job processor. However that can be dangerous practice if it cannot be assured that all requestors are pulling off the same dirty trick. It would just be an added burden to perform such an across the board form of time shifting, and thus Claim 1 literally recites the simpler concept of having the second stamp represent the same time as the first stamp. Moreover, if the scheduler knows that all the requestors are performing such a universal time shifting trick, what in fact is the scheduler scheduling for but the time represented by the second time stamps?)

Applicant's Detailed Reading of LeClair '891

According to LeClair '891, rather than conventionally "pushing" the full bulk of the data that is to be processed (e.g., printed by a shared printer like 135 of Fig. 1) to a remote and centralized spooler (125 of Fig. 1), the bulk of the to-be-processed (i.e. printed) data is kept in a local spool (e.g., 620 of Fig. 6) of the client (or "initiator" 600) and only a "request" for the desired service is sent (pushed) to the central server (610) in the form of a POST command (col. 7, lines 58-59).

LeClair's central server (610) receives the POST command (a.k.a. "request") and analyzes the POST command (URL plus other content described at col. 7, lines 59-67). Irrespective of whether the server directly places the request (the URL) in the server's "queue" (col. 8, line 19) or it first "schedule[es] the request (step 525)" (col. 8, lines 19-20), the URL in the POST command ultimately comes to be placed in the server's "queue" per col. 8, lines 37-45 which explain:

If the request is scheduled, the initiator posts the job to a memory structure or "queue" (step 525). The job posting consists of the URL of the data output file which is stored in a location other than the queue. The server periodically monitors the output queue. Server agent 640 accesses the queue and adds, controls, and removes jobs from the queue. If the queue has jobs for output devices, and the output device is ready (step 530), [*only then does*] the server retrieves the [*bulk*] data from the [*remote*] storage location designated by the URL (step 535).

[Emphasis added and bracketed text in italics also added.]

Therefore, irrespective of what the so-called "scheduling" operation does in LeClair (it is not well defined), the actual processing of the bulk data by the server's output device (680) cannot occur until such time as "the output device is [determined to be] ready (step 530)" **AND** the server has finished retrieving the bulk data (step 535 in Fig. 5) from the remote storage location (from 620 or 622 of Fig. 6). Each of these time durations is a variable because one cannot assure when (or if ever) the output device will ready or how large the bulk data is and therefore when the server will finish retrieving the bulk data (step 535 in Fig. 5). {Note again that paragraph (c) of Claim 1 calls for "storing the received job payload data in the job processor" and that paragraph (d) of Claim 1 refers in the past tense to "the stored payload data" [*Emphasis added.*] Note also that paragraph (b) of Claim 1 calls for "in response to receipt of the first time stamp," [*Emphasis added.*] LeClair's initiator (600) does not respond to receipt of a time stamp. Instead the server 610 "pulls" the bulk data from the storage spool (620 or 622) in response to the server having first looked at the request sitting in the server's queue (step 525 of Fig. 5) and in response to the server finding that the remote output device is ready in step 530. It is the server 610 that "retrieves" or pulls the raw data in step 535 of Fig. 5. The initiator is essentially uninvolved once having "initiated" the process by sending a request to the server.}

Moreover, per the flow at the right side of LeClair Fig. 5, the actual processing of the data by the server's output device (680) --after such data has been "retrieved" (535) by the server and stored in the server-- cannot occur until such time as the server has determined if the received bulk data needs conversion (test step 536), and if yes, until such time as the server has finished performing the conversion (step 537). Additionally, the actual processing of the data by the server's output device (680) cannot occur until after the server finishes

transmitting the converted or raw bulk data to the output device (680) per step 540 of Fig. 5. See col. 8, line 46- col. 9, line 15.

In short:

- LeClair does not send any time stamps;
- LeClair does not schedule when the job (e.g., printing will be performed) but rather when the short-form request will be looked at ("periodically" per col. 8 line 40);
- LeClair's initiator (600) does not respond to any returned time stamps; and
- The time at which LeClair's processing of a job occurs varies depending on numerous variables (i.e. length of data to be fetched from local spool, time when output device is ready, time for converting raw data). There is no definitively scheduled time when the stored data (stored in the server) will be processed.

Incidentally, in arguing all these points, Applicant is not acquiescing to the idea that LeClair is a valid 102(e) reference. To be valid, its original disclosure must meet with all of 35 USC §112 such that it can be argued the PTO would have issued it on the filing date but for some formal delays. (This has to do with the Supreme Court decision which was codified by 35 USC §102(e).) At minimum, LeClair does not describe what "scheduling" entails.

Applicant's Detailed Reading of Phaal '564

Referring to Phaal Fig. 2, a single computer "host" 115 (col. 1, line 10) is coupled to a plurality of personal computers 113/119. Inside the host 115 there is a deferral manager 131, a scheduler 135 and a plurality of processing servers 117, 118.

The clients (PC's 113/119) send "messages" to the host. The host has to determine how to prioritize the admission and processing of these messages (e.g., HTTP download requests -- col. 1, line 23). There are no separate entities identified as payloads that are sent from the

clients 113/119 to the host 115. The transmissions from clients 113/119 to host 115 are referred to as "messages".

When too many messages are thrown at the host, it refuses "admission" to some of them. Phaal is directed to determining which messages will be refused admission and what to do when a message is refused admission.

Per one embodiment of Phaal a "deferral manager [131]... formulates a time indication which tells a client system [119] when it [119] can expect to gain admission to the host [115]" if it resubmits its message at such later time (col. 3, line 8, *Emphasis added, bracketed text added*). In doing so, the deferral manager is not promising the client a specific time when it will be best to resubmit its message for guaranteed admission, but rather, the deferral manager is estimating when it expects to have resources freed up or incoming message traffic reduced such that the turned away client will have a better chance of getting its prioritized message "admitted" into the service queue of the host. Admission is not the same as job processing though in Phaal.

Referring to Fig. 1 of Phaal, an "admitted" message can be left to sit in a front end reordering message queue 28 or a not-yet-in-progress queue 21 (see col. 4, lines 65-etc. and col. 5, line 33). Thus admission does not indicate when the message's job will be processed, but rather when it will be dropped into a competitive queue for servicing according to priority.

The Office action points to Phaal col. 6, lines 59-etc. This part is discussing what happens when a client has been refused admission. The deferral manager is basically telling the client to try again at a later time, but it is not guaranteeing admission or a specific time of service. **More specifically, Phaal col. 6, lines 50-etc. state:**

Since the admission control gateway 125 defers some messages ... when resources are stretched, it is desired to ... in order that the user of the client system will not become frustrated or continually re-submit the message (thereby further overloading server resources). To accomplish this end, the admission control system 111 further includes a deferral manager 131, which formats and provides a response message 133 to the client system ... Preferably, the deferral manager 131 is coupled to a scheduler 135 which, together with the deferral manager, calculates a later time when it can be expected that the deferred message can be processed by the server 117. ... For example, the scheduler can compile statistics based on day-to-day operation of the server and times when the processing resources of the server tend to be less strained; in this example, the scheduler could determine that a particular server is "less busy" from twelve O'clock noon until one O'clock P.M., and could defer a

client system until twelve O'clock noon and the one hour time range thereafter. Alternatively, the scheduler could simply set "appointments" (e.g., two for every five minutes) and simply return to the deferral manager 131 a time for the next available appointment. In the preferred embodiment, the scheduler uses the latter function and defers messages for a minimum predetermined amount of time, e.g., 300 seconds as indicated by Table I, below; in conjunction with a time set by a web page which is downloaded to the client, the client's message is later accepted on a priority basis if the client contacts the server within a defined interval following the time. Implementation of the scheduler is effected in the preferred embodiment via software.

A number of mechanisms can also be implemented such that the admission control system 111 recognizes a deferred message as a priority message following re-submission. In the preferred embodiment, the deferral manager 131 generates a "key" in the form of a "cookie" which the admission control system writes to memory of the client system. ... This web page visually displays to a user of the client system an informative text message, e.g.,

"We're sorry, but our server is temporarily serving other clients; to better assist you, we have scheduled an appointment for your transaction, and if you do not exit this web page, your browser will automatically contact us in 23 seconds."

[Emphasis added. Some text skipped to reduce clutter.]

So to summarize, Phaal does not schedule a specific time when it guarantees service to a given message (i.e., request) but rather Phaal merely estimates when it might be a better time for the message to come knocking back on the admission manager's door. That still does not guarantee admission because a much higher priority message might unexpectedly arrive at the same time and the once-refused message may be turned away again and told to come back at yet a later time. Accordingly, the processing of its job has not been "scheduled" in the ordinary sense of the word. Additionally, Phaal does not teach or suggest that the clients send payloads with second time stamps attached in response to having received something that qualifies as a first time stamp.

In view of this, it is respectfully submitted that the finding of fact at the bottom of OA page 4 is in error. Phaal's scheduler 135 does not schedule jobs. Instead it "estimates" a later time span (interval) when the refused-message should come knocking back again on the admission manager's door for a better chance of getting admitted. Admission alone does not mean that the request (message) will be serviced at the time of admission.

In view of the above detailed reviews of both of LeClair and Phaal, it is respectfully submitted that the finding of fact at the top of OA page 5 is in error. Neither of LeClair's server 640 or Phaal's scheduler 135 schedules a time when a job is promised to be performed. LeClair's system schedules when it will look at "requests" in its request queue and at that time the payload (i.e., data to be printed) is not yet stored in the server. Phaal's scheduler 135 merely estimates what future time (as measured by whom?) is a good time for the client to resubmit a message for possible, but not guaranteed, "admission".

No reasonable basis for combining articulated

It is respectfully submitted that no reasonable basis is articulated at the top of OA page 5 for specifically choosing LeClair and Phaal and combining them. "Scheduling" and "computers" are too broad of a category to justify picking specifically Phaal for combination with LeClair. In LeClair, the requestor (initiator) stores a bulk of data for uploading to the central server and the server "pulls" it in when the server is good and ready. So the requestor does not respond to receipt of a time stamp. In LeClair, requests are not refused "admission". Therefore there is no motivation for looking to Phaal.

By contrast, in Phaal, there is no bulk payload being pushed towards the central server. Instead, it is generally the server (e.g., web server) that will be downloading massive data to the clients (individual PC's out there on the web that pull in the downloaded data) and the problem addressed is that of the web server sometimes being overwhelmed with too many incoming "messages" (requests) due to unpredictable web traffic. So Phaal's server tells some messages to try and come back at a later time. But Phaal's server does not promise that when they do come back at the later time they will be serviced, just that they have a better chance of being "admitted" if they so cooperate.

Even if, assuming arguendo, LeClair and Phaal could be properly combined; they would nonetheless fail to produce the subject matter of Claim 1 because neither schedules a specific time in the time frame of the server when the job will be done.

Claims 41-42 compared against LeClair and Phaal

With regard to Claim 41 (as amended above), it has been implicitly shown above that neither of LeClair and Phaal sends something that qualifies as "(b) ... corresponding grant signals, where the grant signals include one or more time stamps indicating a time point with a timing frame of processing part of the system that *the processing part is scheduled to process the data* sourced by the data-sourcing circuit." [*Emphasis added.*] There is no scheduled time in LeClair at which the server (610) or output device (680) is scheduled to process the bulk data (e.g., print data) supplied by the initiator (600). There is no scheduled time in Phaal at which its web server (115) is scheduled to process the message supplied by a client (113/119). Phaal's "appointments" merely indicate a better time when the messages should have a better chance of gaining admission if they come knocking in the indicated interval.

The above amendment to Claim 41 is not in response to the applied art but rather to correct an overlooked language problem.

With regard to original Claim 42, Applicant concedes that adding a new PC to a network may at times "include" the adding of circuit boards. However, the new PC will have many boards (i.e., network interface board, hard drive board in the hard drive, etc.). It will not inherently be a single board.

Claim 43 compared against LeClair and Phaal

Unlike Claim 41, claim 43 considers the case where the system is expanded on the job-processor side rather than the requestor side. That would be akin to adding a new server into the system of LeClair or Phaal. However, neither of LeClair or Phaal remotely suggests such an expansion.

Claim 17

Please note that the preamble of Claim 17 recites "wherein for each of the alignment queues" [*Emphasis added*] just as does the preamble of Claim 16 from which 17 depends. It is believed that there is no ambiguity given that it is discussing the "each" of the alignment

queues. However, if the Examiner has alternative language to propose, Applicant is open to hearing such a suggestion.

Similarly, with regard to the "granted, processing time slot", Claim 17 depends from Claim 16 and in the latter claim, it is explained "(b.1) ... so that the popped payload signal can be duly processed during the granted, processing time slot ". It is believed that the preamble language of Claim 17, "wherein for each of the alignment queues" [*Emphasis added*] indicates that indeed it is the same granted, processing time slot as that of Claim 16. However, if the Examiner has alternative language to propose, Applicant is open to hearing such a suggestion.

CONCLUSION

In light of the foregoing, Applicant respectfully submits that the outstanding rejections have been overcome. Should any negative action be contemplated by the Examiner, it is respectfully requested that he contact the undersigned at (408) 392-9250 to discuss the application.

The Commissioner is authorized to charge any underpayment or credit any overpayment to Deposit Account No. 50-2257 for any matter in connection with this response, including any fee for extension of time and/or fee for additional claims, which may be required.

I hereby certify that this correspondence is being deposited with the United States Postal Service as First Class Mail in an envelope addressed to: Commissioner for Patents, P.O. Box 1450, Alexandria, VA 22313-1450, on July 27, 2006.

 7/27/06

Attorney for Applicant(s)

Date of Signature

Respectfully submitted,



Gideon Gimlan
Attorney for Applicants
Reg. No. 31,955

MacPherson Kwok Chen & Heid
LLP
1762 Technology Drive,
Suite 226
San Jose, CA 95110
Telephone: (408) 392-9250
Facsimile: (408) 392-9262

MacPherson Kwok Chen & Heid LLP
1762 Technology Drive, Suite # 226
San Jose, CA 95110
Tel: (408) 392-9250